

①9 RÉPUBLIQUE FRANÇAISE
INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE
PARIS

①1 N° de publication :
(à n'utiliser que pour les
commandes de reproduction)

2 757 664

②1 N° d'enregistrement national : 96 15997

⑤1 Int Cl⁶ : G 07 F 7/08, G 06 K 19/073

⑫

DEMANDE DE BREVET D'INVENTION

A1

②2 Date de dépôt : 24.12.96.

③0 Priorité :

④3 Date de la mise à disposition du public de la
demande : 26.06.98 Bulletin 98/26.

⑤6 Liste des documents cités dans le rapport de
recherche préliminaire : *Se reporter à la fin du
présent fascicule.*

⑥0 Références à d'autres documents nationaux
apparentés :

⑦1 Demandeur(s) : CP8 TRANSAC — FR.

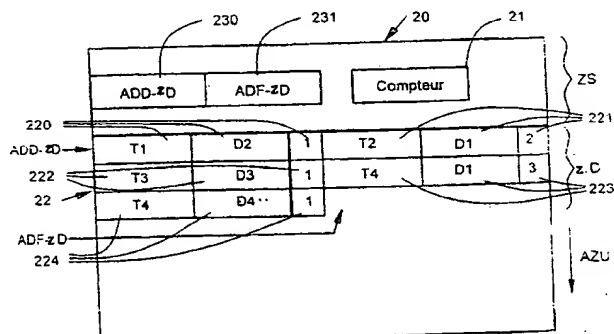
⑦2 Inventeur(s) : LAPIE RONAN.

⑦3 Titulaire(s) :

⑦4 Mandataire : CABINET DEBAY.

⑤4 TERMINAL ET PROCEDE D'AUTODIAGNOSTIC OU DE SUPERVISION ET OBJET PORTATIF UTILISE DANS
UN TEL TERMINAL OU PROCEDE.

⑤7 Terminal doté d'un programme d'application; d'au
moins une sortie constitué soit par un affichage, soit par
une imprimante, soit par un réseau de communication, soit
par un objet portable et coopérant avec un objet portable
doté d'une zone de mémoire non-volatile (ZD) contenant
des données et comportant un lecteur communiquant avec
ledit objet portable, caractérisé en ce que l'appareil com-
porte des moyens de lecture ou de stockage dans sa mé-
moire de données (Ti, Dj, Sk) d'autodiagnostic ou de su-
pervision et un moyen d'émission desdites données vers
des sorties (1-4) spécifiées en fonction d'informations four-
nies par les données d'autodiagnostic ou de supervision
suite à l'exécution d'au moins une tâche (Tt) de son pro-
gramme d'application en relation avec l'objet portable.



FR 2 757 664 - A1



**TERMINAL ET PROCEDE D'AUTODIAGNOSTIC OU DE SUPERVISION
ET OBJET PORTATIF UTILISE DANS UN TEL TERMINAL OU PROCEDE**

La présente invention concerne un terminal et un procédé d'autodiagnostic ou de supervision et l'objet portatif de type carte à microcircuit utilisé dans un tel procédé ou terminal ou lecteur. Ces objets comportent une Unité Centrale, une mémoire programme contenant du code exécutable :
5 constituant le système d'exploitation, une mémoire de donnée non volatile et programmable et, une ou plusieurs interfaces de communication. Les terminaux sont des appareils dotés d'une interface compatible avec celles
10 des objets, ils possèdent une Unité centrale et un logiciel capable de communiquer et d'exploiter les données provenant de la mémoire non volatile de l'objet portable.

En général, les terminaux sont équipés d'un logiciel spécifique correspondant à leur utilisation, par exemple, les terminaux portables de
15 paiement sont dotés d'un programme d'exploitation de type bancaire. Ce logiciel est réalisé ou spécifié par l'organisme qui gère cette application, dans l'exemple cité, c'est un organisme bancaire. Cet organisme n'est généralement pas le fabricant du terminal, il achète ou fait fabriquer la partie matérielle c'est-à-dire le terminal, il y introduit le programme spécifique afin
20 de configurer son terminal pour sa propre application. L'organisme bancaire y trouve son avantage en achetant un produit standard donc bon marché, et en l'adaptant selon ses besoins. Le fabricant propose un modèle de base qui peut convenir à plusieurs applications, ce qui lui permet d'étendre son marché.

25 L'organisme exploitant une application donnée peut désirer disposer de plusieurs modèles de terminaux lecteurs de cartes, il n'est pas souhaitable de développer des logiciels applicatifs pour chaque terminaux. Ce qui a conduit les fabricants à implémenter une couche logicielle de base qui assure l'interface entre le matériel et le logiciel applicatif. Cette couche
30 logicielle assure l'adaptation entre un même logiciel applicatif et des terminaux différents. Une façon de faire consiste à créer un interpréteur, de telle sorte que l'organisme peut développer son application en langage évolué bien connu et ceci quasiment indépendamment des contraintes du matériel. Une autre façon de faire est d'organiser une couche logicielle de

bas niveau qui gère toutes les entrées sorties matérielles et, à mettre à la disposition de l'organisme exploitant une bibliothèque de primitives que le logiciel applicatif va appeler.

Dans tous les cas, on doit être en mesure de valider ou de tester le terminal
5 dans son ensemble. La validation ou le test du terminal doit prendre en compte les deux parties : le matériel avec son logiciel de base et le logiciel d'application. Un auto-test permet de vérifier chaque équipement du terminal, il est généralement constitué d'une routine implémentée dans le logiciel de base. Le test du logiciel d'application doit être réalisé en
10 laboratoire, il est important dans ce type d'application de bien vérifier le fonctionnement du programme avant sa mise en service. Toutefois, la multiplicité des cartes entraîne un très grand nombre de cas particuliers non reproductibles en laboratoire. L'invention suivante a pour but de valider ou tester en conditions normales d'utilisation le fonctionnement d'un logiciel
15 d'application.

A cet effet, l'invention concerne un terminal doté d'un programme d'application, d'au moins une sortie constituée soit par un affichage, soit par une imprimante, soit par un réseau de communication, soit par un objet portable et coopérant avec un objet portable doté d'une zone de mémoire
20 non-volatile contenant des données et comportant un lecteur communiquant avec ledit objet portable, caractérisé en ce que l'appareil comporte des moyens de lecture ou de stockage dans sa mémoire de données de diagnostic ou de supervision et un moyen d'émission desdites données vers des sorties spécifiées en fonction d'informations fournies par les données
25 d'autodiagnostic ou de supervision suite à l'exécution d'au moins une tâche de son programme d'application en relation avec l'objet portable.

Selon une autre caractéristique, les moyens d'émission des données d'autodiagnostic sont activés un certain nombre de fois.

Selon une autre caractéristique, les moyens d'émission des données
30 d'autodiagnostic ou de supervision comporte des moyens d'écriture dans l'objet portable connecté à l'appareil.

Selon une autre caractéristique, les données d'autodiagnostic ou de supervision sont constituées par au moins un triplet d'information

correspondant pour une première information à une tâche déterminée du programme d'application, pour la seconde information à un type de données corrélées à la tâche exécutée et à présenter à une sortie, et pour la troisième information à une valeur de spécification de la sortie à laquelle le type de données doit être présenté parmi celles présentes sur le terminal.

Selon une autre caractéristique, l'appareil dispose d'un moyen de test de la présence de données d'autodiagnostic ou de supervision dans un objet portable et de déclencher la lecture et le stockage de ces données dans une zone spécifique ZTD de la mémoire du terminal.

10 Selon une autre caractéristique, le terminal comporte des moyens d'introduction de données d'autodiagnostic ou de supervision dans un objet portable.

Un autre but de l'invention est de proposer un procédé de supervision du fonctionnement d'un terminal ou d'autodiagnostic d'un terminal.

15 Ce but est atteint par le fait que le procédé d'autodiagnostic ou de supervision à partir d'au moins un triplet d'information correspondant pour une première information à une tâche déterminée d'un programme d'application exécuté soit par un objet portable soit par un terminal, pour la seconde information à un type de données corrélées à la tâche exécutée et
20 à présenter sur une sortie et pour la troisième information à une valeur de spécification de la sortie parmi celles présentes sur le terminal se caractérise en ce qu'il consiste :

- à exécuter une tâche du programme d'application sur le terminal ;
- à tester un indicateur soit dans le terminal soit dans l'objet portable pour
25 déterminer si une fonction d'autodiagnostic ou de supervision est opérationnelle, puis dans le cas d'une réponse positive ;
- à rechercher dans la mémoire soit de l'objet portable soit du terminal s'il existe parmi les triplets d'information stockés un triplet dont la première information correspond à la tâche déterminée exécutée par le terminal ou la
30 carte ;

- à émettre vers la sortie spécifiée par le triplet ainsi lu la valeur de la donnée corrélée à la tâche exécutée et à référencer par la seconde information du triplet.

5 Selon une autre particularité, le procédé comporte une étape de test consistant à déterminer s'il existe d'autres tâches à exécuter et à la suite de l'exécution de ces tâches à rechercher tous les triplets d'information correspondants à l'exécution de cette tâche.

10 Selon une autre particularité, le procédé comporte une étape de lecture d'un objet portable mémorisant dans sa mémoire non-volatile une pluralité de triplets et une étape de stockage de ces triplets dans une zone de mémoire non-volatile du terminal suivie d'une étape d'activation d'un indicateur de fonction d'autodiagnostic ou de supervision active.

15 Selon une autre particularité, le procédé comporte une étape de test pour déterminer si l'objet portable est une carte spécifique à la fonction d'autodiagnostic ou de supervision ou une carte dite banale.

20 Selon une autre particularité, les données d'autodiagnostic ou de supervision sont constituées par un quatrième champ d'information contenant, au départ dans l'objet portable l'adresse d'écriture (Adr-V), le nombre d'octets à écrire (Nb-V), après l'opération d'autodiagnostic la valeur à écrire (Val).

Un autre but de l'invention est de proposer un objet portable utilisable avec le terminal et le procédé d'autodiagnostic ou de supervision.

25 Ce but est atteint par le fait que l'objet portable est une carte à microprocesseur fonctionnant grâce à un système d'exploitation mémorisé sur la carte et comportant une mémoire non-volatile contenant au moins un triplet d'information dans une zone déterminée de cette mémoire non-volatile dont la position est définie par des champs d'adresse situés dans la partie de mémoire servant à stocker le système d'exploitation.

30 Selon une autre particularité, la partie de mémoire non-volatile servant à mémoriser le système d'exploitation comporte également dans un champ

mémoire une information constituant un compteur d'utilisation de la fonction d'autodiagnostic.

Selon une autre particularité, la zone mémoire stockant le système d'exploitation comporte un champ permettant de mémoriser un indicateur
5 d'activation de la fonction d'autodiagnostic ou de supervision.

D'autres particularités et avantages de la présente invention apparaîtront plus clairement à la lecture de la description ci-après faite en référence aux dessins annexés dans lesquels :

la figure 1 est un tableau explicatif des triplets constitués des tâches
10 élémentaires, des types de données et des types de sorties que l'on peut associer à l'exécution de chacune des tâches d'un programme d'application ;

la figure 2 est un schéma des zones mémoire non-volatile utilisées sur un objet portable de test nécessaire à la mise en oeuvre du procédé de l'invention ;

15 la figure 3 représente les différentes étapes d'exécution des programmes d'initialisation du terminal et d'exécution de la fonction d'autodiagnostic ;

la figure 4 représente une vue schématique des zones de stockage des informations dans la mémoire non-volatile d'un objet portable dit banal ;

la figure 5 représente les différentes étapes d'exécution d'un programme
20 d'initialisation du terminal et d'exécution de la fonction d'autodiagnostic sur ce terminal avec une carte dite banale.

Une manière de réaliser l'invention consiste dans une première variante, d'une part à utiliser une carte à microprocesseur fonctionnant grâce à un système d'exploitation mémorisé sur la carte et constituant une "carte à puce
25 dite intelligente" qui est préalablement initialisée avec des données d'autodiagnostic, d'autre part à faire prendre en compte ces données par le terminal dont on veut tester le logiciel d'application.

Un logiciel d'application dans un terminal peut se décomposer en tâches élémentaires qui se succèdent à des moments déterminés. Par exemple,
30 pour une application bancaire, on peut décomposer une transaction par les

tâches élémentaires (Ti) suivantes : Vérification (T1, Fig. 1) si la carte introduite est habilitée, Authentification (T2) du porteur, Acquisition (T3) des données de la transaction dans le terminal, Inscription de ces données dans le microcircuit de la carte, Validation (T4) de la transaction dans le terminal et la carte.

D'autres part et toujours lors d'une transaction, le logiciel d'application manipule des données, ces données peuvent être utilisées temporairement comme, d'une part le Code élaboré par le porteur (Cp) qui est stocké dans la mémoire du terminal, d'autre part l'identité du porteur (Ip) qui est stockée dans la carte ou encore comme le Montant de la transaction (Mt) ou la Date de la transaction (Dt) qui sont stockés dans le terminal et la carte. Lors de l'exécution de chaque tâche élémentaire, chacune de ces données peuvent être initialisées, modifiées ou inchangées. La fonction d'autodiagnostic d'utilisation du logiciel d'application consiste à contrôler les données de la transaction au moment de certaines tâches et ceci dans des conditions normales d'utilisation. Cette fonction peut être réalisée soit par le logiciel de base, soit par le logiciel d'application ;

Pour ce faire, l'opérateur chargé du contrôle élabore une grille composée d'un coté des tâches élémentaires identifiables notées Ti et de l'autre coté, des données Dj constituées par les informations telles que : Cp, Ip, Mt et Dt. La figure 1 montre un exemple d'une telle grille. Pour contrôler le déroulement correct d'une transaction, l'opérateur choisit de vérifier la valeur de certaines données lors de l'exécution de tâches bien précises. Cela revient à associer une donnée Dj à une tâche Ti, ces associations sont symbolisées par des croix sur la grille de la figure 1. Un troisième élément d'information Sk est rajouté. La valeur de ce code précise le type de sortie utilisée pour émettre la donnée à contrôler : vers le réseau lorsque Sk est à une première valeur par exemple "1" (Sk = 1), vers l'imprimante lorsque Sk est à une deuxième valeur par exemple "2" (Sk = 2), ou vers l'écran lorsque Sk est à une troisième valeur par exemple "3" (Sk = 3). L'opérateur introduit les triplets d'information (Ti, Dj, Sk) dans une unité centrale spéciale dite de diagnostic, cette unité centrale est dotée d'un lecteur de carte. Le logiciel de diagnostic est configuré selon l'application à tester de telle sorte que les triplets (Ti, Dj, Sk) sont identifiés lors de la saisie sur écran par l'indication

précise des tâches élémentaires et des données à contrôler et non les références numériques Ti, Dj, Sk...

La carte contenant les données d'autodiagnostic est : soit une carte spéciale, soit une carte banale utilisée couramment pour une application.
5 Une description détaillée d'une réalisation est donnée pour chaque cas.

Décrivons tout d'abord le cas où une carte spéciale dite "carte de test" (20, Fig. 2) est utilisée pour contenir les données d'autodiagnostic. Une procédure sécuritaire est mise en oeuvre pour éviter qu'un fraudeur puisse se servir d'une telle carte de façon non autorisée. La carte de test contient,
10 dans la zone mémoire secrète non représentée, un code secret dit de diagnostic "KD". Ce code secret doit être préalablement présenté à la carte qui le vérifie et, s'il est égal à un code de référence, autorise l'écriture des données d'autodiagnostic dans la mémoire programmable de la carte.

En prévision du stockage des données d'autodiagnostic, la mémoire
15 programmable non volatile de la carte de test possède en plus de la zone système ZS contenant le système d'exploitation de la carte et de la zone (AZU autre zone utilisable) permettant d'autres mémorisations, une zone (22) appelée "ZD". C'est dans cette zone que sont stockés successivement les triplets (Ti, Dj, Sk). Ainsi une première zone (220) d'une mémoire permet
20 le stockage du premier triplet T1, D2, 1 ; une deuxième zone (221) permet le stockage du deuxième triplet T2, D1, 2 ; une troisième zone (222) permet le stockage du troisième triplet T3, D3, 1 ; une quatrième zone (223) permet le stockage du quatrième triplet D4, D3, 3 ; une cinquième zone (224) permet le stockage du triplet T4, D4, 1 ; T1, T2, T3, T4, D1, D2, D3, D4 représentant
25 respectivement les informations de la figure 1. Il est bien évident que l'objet portable peut comporter plus ou moins de triplet selon le type de supervision ou d'autodiagnostic que l'on veut effectuer sur les tâches exécutées par le programme d'application. La zone ZD est repérée par son adresse de début : "ADD_ZD" et son adresse de fin "ADF-ZD", les deux valeurs d'adresse sont
30 stockées dans la partie (230, 231) de la mémoire programmable allouée au système d'exploitation.

La mémoire programmable non volatile est de type EPROM, EEPROM, FeRAM, SRAM ou FLASH. La figure 2 décrit l'organisation de cette mémoire

en reprenant les informations citées par la figure 1. Avantageusement, la donnée Dj est l'adresse physique de la donnée à contrôler dans la mémoire de travail du terminal.

Une fois programmée, la carte de test est introduite dans le terminal dans lequel doit se dérouler la fonction d'autodiagnostic. La figure 3 est un organigramme illustrant la chronologie des événements du programme constituée d'une séquence d'attente et de test (1, 2, 3), le test déclenchant en fonction du résultat soit une séquence de chargement (4 à 7, Fig. 3) du terminal avec les données d'autodiagnostic, soit une séquence d'exécution du programme d'autodiagnostic (8 à 16, Fig. 3) qui est incorporé soit dans le logiciel de base du terminal, soit dans le logiciel d'application. L'étape 1 est l'initialisation du terminal après une mise sous tension et l'étape 2 est la phase d'attente d'un ordre ou d'une introduction de carte. A l'étape 3, le terminal teste si la carte introduite dans le lecteur est une carte banale, et à l'étape 4 si la carte est une carte de test. Dans ce dernier cas, le terminal exécute à l'étape 5 une procédure d'authentification de la carte par un code de référence ou, par un schéma classique d'authentification challenge-réponse utilisant un algorithme et une clé secrète (KD).

Une fois la carte de test identifiée et authentifiée, le programme du terminal lit à l'étape 6 les informations contenues dans la zone ZD. La sélection et la localisation des triplets s'effectuent à l'aide des deux pointeurs ADD_ZD et ADF-ZD. Les triplets (Ti, Dj, Sk) successivement lus dans la zone ZD sont stockés dans le même ordre dans une zone de la mémoire du terminal appelée ZTD. Une fois le dernier triplet stocké dans la zone ZTD, le programme du terminal, à l'étape 7, positionne à l'état actif un indicateur d'autodiagnostic "Ind_DT" dans la mémoire du terminal. Puis le programme du terminal reboucle en attente d'une commande ou d'une autre introduction de carte sur l'étape 2.

Une nouvelle carte est introduite, c'est une carte banale, compatible avec l'application traitée par le terminal. On a dit précédemment que le logiciel d'application dans le terminal se décompose en tâches élémentaires Tt que l'on peut individuellement exécuter (étape 8). A la fin de l'exécution de chaque tâche que l'on peut référencer par le code Tt, le programme d'application teste à l'étape 9 l'indicateur Ind_DT du terminal. S'il est inactif,

la fonction d'autodiagnostic n'est pas opérationnelle, le programme continue à exécuter les autres tâches. Si l'indicateur Ind_DT est actif, le programme du terminal recherche à l'étape 10 dans la zone ZTD de la mémoire du terminal le premier triplet (Ti, Dj, Sk) pour lequel Tt = Ti c'est à dire, s'il y a
5 une donnée à tester à la suite de la tâche qui vient de s'exécuter. Si oui, à l'étape 11, la valeur "Val" de cette donnée (Dj) est stockée temporairement dans la mémoire du terminal et en fonction de la valeur de Sk, elle est traitée de la façon suivante :

Si Sk est égal à "1" (étape 12), la donnée Dj doit être envoyée vers le
10 réseau. Un bloc de trois données est alors constitué : la valeur du champ Tt, la référence Dj de la donnée à analyser et la valeur "Val" de cette donnée extraite de la mémoire du terminal. Ces blocs sont mémorisés les uns à la suite des autres dans une zone de la mémoire du terminal appelée "ZDR". Le contenu de cette zone est envoyé vers le réseau à la fin de la transaction
15 ou lors d'une demande par le réseau des données d'autodiagnostic. Une fois toutes les données transmises, la zone ZDR est vidée pour être réutilisée lors d'une nouvelle introduction de carte.

Si Sk est égal à "2" (étape 13), la donnée Dj doit être envoyée vers l'imprimante du terminal pour impression. Un message est alors créé dans le
20 tampon logiciel de l'imprimante, il est composé d'un texte (code ASCII) rappelant la nature de la donnée, par exemple "MONTANT : "suivi de la valeur décimale ou hexadécimale de la donnée Dj, le message se termine par un séparateur et un "RETOUR CHARIOT - SAUT DE LIGNE". On peut regrouper tous les messages d'autodiagnostic et les imprimer à la fin de la
25 transaction.

Si Sk est égal à "3" (étape 14), la donnée doit être envoyée vers l'afficheur du terminal pour affichage. Un message est alors créé dans le tampon de l'afficheur, composé d'un texte (code ASCII) rappelant la nature de la donnée par exemple "MONTANT : "suivie de la valeur décimale ou
30 hexadécimale de la donnée Dj. Les messages correspondant à chaque élément (Tt, Dj, "3") sont affichés successivement pendant un certain temps fixé par le programme. On peut regrouper tous les messages et les afficher à la fin de la transaction, le défilement des messages peut être contrôlé par l'appui sur une touche du clavier du terminal.

Une fois la donnée Dj traitée, le programme vérifie à l'étape 15, s'il existe dans ZTD d'autres triplets pour lesquels $T_t = T_i$. Si oui, le programme reboucle à l'étape 11 et traite un nouveau triplet. Pour chaque tâche élémentaire, la recherche de triplets s'effectue en balayant toute la zone ZTD. S'il ne reste plus de triplets ($T_i = T_t$, Dj, Sk) à traiter, le programme, à l'étape 16, continue en séquence et passe éventuellement dans une variante à une autre tâche sans exécuter les étapes 17 et 18 décrites ci-après. S'il n'y a plus d'autres tâches à exécuter, le programme reboucle à l'étape 3, en attente d'une commande ou d'une nouvelle introduction de carte.

10 On peut associer un compteur initialisé avec un certain nombre à l'indicateur Ind-DT dans le terminal de telle sorte que, la fonction d'autodiagnostic est activée que pour ce nombre d'introduction de carte banale. Pour ce faire, l'opérateur a préalablement introduit ce nombre dans un emplacement spécifique (21, fig. 2) de la mémoire programmable de la carte de test par exemple, à côté des emplacements (230, 231) de ADD_ZD et ADF-ZD. Ce
15 nombre est dans ce cas stocké dans la mémoire du terminal après l'introduction de la carte de test, à l'étape 6. Puis ce nombre est décrémenté (étape 17) à chaque fin d'exécution d'une fonction d'autodiagnostic, (sortie OUI de l'étape 16). Lorsque celui-ci passe à "0", l'indicateur Ind_DT est positionné inactif (étape 18) et, éventuellement, le contenu de la zone ZTD
20 est effacée.

Si ce compteur n'est pas installé les étapes 17 et 18 n'existent pas et la fonction d'autodiagnostic s'exécute une seule fois ou, indéfiniment jusqu'à ce qu'une nouvelle introduction de la carte de test rebascule l'indicateur Ind_DT
25 en position inactive.

Il est possible d'éviter l'utilisation d'une carte de test et d'employer que des cartes banales à condition qu'elles supportent les fonctions spéciales d'autodiagnostic. Pour cela la mémoire programmable de la carte banale contient en plus de la zone système ZS et de la zone utilisateur ZU, une
30 zone ZD qui est repérée par son adresse de début : "ADD_ZD" et son adresse de fin "ADF-ZD" (voir figure 4). La mémoire programmable de la carte banale contient dans sa zone système également à un emplacement (232) un indicateur "Ind_D" signalant si la fonction d'autodiagnostic est active ou non. Toutes ces données ADD-ZD, ADF-ZD, Ind-D sont stockées

dans des emplacements (230, 231, 232), de la partie ZS de la mémoire programmable allouée au système d'exploitation. Les deux valeurs d'adresse sont déterminées et écrites lors de la personnalisation de la carte la zone ZD, cette méthode est simple à mettre en oeuvre mais possède
5 l'inconvénient de nécessiter la réservation d'un emplacement important dans toutes les cartes pouvant servir pour l'autodiagnostic.

Avantageusement, l'emplacement de la zone ZD peut être alloué dynamiquement par le système d'exploitation de la carte à la suite de la bonne présentation du code KD. L'opérateur précise alors à la carte le
10 nombre de triplets (Ti, Dj, Sk) ou le nombre d'octets à réserver pour ZD. Le système d'exploitation de la carte recherche alors dans la mémoire programmable un emplacement vierge suffisant. Si la mémoire ne contient pas un tel emplacement vierge, le système d'exploitation renvoie un message d'erreur et interrompt la procédure d'introduction des données
15 d'autodiagnostic. Dans le cas contraire, la place est suffisante le système d'exploitation mémorise l'adresse de début : "ADD_ZD" et l'adresse de fin "ADF-ZD". Nous verrons par la suite comment, après l'exécution de la fonction d'autodiagnostic, on peut effacer la présence de la zone ZD et ainsi libérer cet emplacement mémoire.

20 De la même façon que pour la carte de test. Une procédure sécuritaire est prévue pour éviter qu'un fraudeur puisse se servir d'une carte banale pour y introduire des données d'autodiagnostic. Un mécanisme de type challenge-réponse avec algorithme et code secret permet d'authentifier l'opérateur et d'autoriser l'écriture et la lecture (nous le verrons pourquoi par la suite) des
25 triplets dans ZD.

Dans le cas d'une carte banale utilisée pour transmettre les données d'autodiagnostic, le code Sk peut prendre une quatrième valeur : 4, cette valeur indique que la valeur de la donnée Dj à contrôler est inscrite dans la carte. Un quatrième champ situé à une adresse "Adr-V" est alors alloué à la
30 suite du triplet (Ti, Dj, Sk=4) et on mémorise donc des quadruplets. La taille de ce champ correspond à celle de la donnée à inscrire, l'opérateur doit donc préciser le nombre d'octets "Nb-V" de ce quatrième champ et son contenu est constitué au départ de l'adresse (Adr-V) d'écriture puis comme on le verra par la suite après la sortie par la valeur "Val". Le cinquième

triplet (225) de la figure 4 possède cette structure. Lorsque tous les triplets (Ti, Dj, Sk) (220 à 225) sont introduits dans la zone ZD, l'indicateur "Ind_D" est positionné actif signalant ainsi que la fonction d'autodiagnostic est active dans cette carte.

- 5 La figure 5 montre le déroulement des opérations lorsque la carte décrite précédemment est introduite dans un terminal. L'étape 1 est l'initialisation du terminal après une mise sous tension et l'étape 2 est la phase d'attente d'une introduction de carte, le programme continue, lorsque la carte est reconnue comme compatible avec l'application par reconnaissance par le
- 10 terminal de la présence des informations nécessaires. A cette étape 2, le programme effectue la sélection de l'entité correspondant à l'application. A la différence de la carte de test, lorsque la carte banale est introduite par le porteur, ce dernier peut parfaitement ignorer que la fonction d'autodiagnostic est active.
- 15 A l'étape 3, le terminal teste si l'indicateur Ind_D dans la carte est positionné actif et donc, si la fonction d'autodiagnostic est opérationnelle. L'indicateur peut être émis soit par une valeur particulière dans les octets émis par la carte lors de la mise sous tension, soit par une valeur particulière émise lors de la sélection de l'entité correspondant à
- 20 l'application utilisée sur la carte. Si Ind_D est actif, le programme passe à l'étape 4. Au cours de cette étape 4, la zone ZD est lue à l'aide des deux valeurs d'adresse ADD-ZD et ADF_ZD et l'ensemble des triplets lus dans la carte sont stockés dans la mémoire ZTD du terminal. Si le triplet comporte une information Sk dont la valeur est "4", le système d'exploitation de la
- 25 carte renvoie en plus des trois valeurs Ti, Dj et Sk, l'adresse "Adr-v" du quatrième champ réservé pour inscrire la donnée dans ZD et le nombre d'octet "Nb_v" de ce champ. Pour des raisons de sécurité, l'accès en lecture à la zone ZD de la carte n'est accordé par le système d'exploitation de la carte que si l'indicateur Ind_D de la carte est actif. Une fois toutes les
- 30 informations contenues dans la zone ZD stockées dans la zone mémoire ZTD du terminal, le terminal positionne son indicateur d'autodiagnostic Ind-D en position active (voir étape 5 de la figure 5). Les étapes 3, 4 et 5 sont des parties de la séquence d'initialisation du dialogue entre la carte banale et le terminal et s'exécutent avant l'exécution du programme d'application.

On a dit précédemment que le logiciel d'application se décompose en tâches élémentaires que l'on peut individuellement tester. A la fin de l'exécution de chaque tâche que l'on peut référencer par un numéro Tt (étape 6), par exemple le logiciel de base, reprend la main et teste si l'indicateur de diagnostic interne au terminal est actif (étape 7). Si celui-ci est actif, à l'étape 8, le programme recherche s'il existe un élément (Ti, Dj, Sk) stocké dans la zone ZTD qui possède une valeur de champ Ti égale à celle de Tt. Si oui : il y a une donnée (référéncé Dj) à contrôler à la suite de la tâche qui vient de s'exécuter, la valeur de cette donnée est alors stockée temporairement dans la mémoire du terminal. En fonction de la valeur de Sk, elle est traitée de la façon suivante (étape 9) :

Si Sk est égal à "1" (étape 10), la donnée Dj doit être envoyée par le terminal vers le réseau. Un bloc de trois données est alors constitué de la valeur du champ Tt, de la référence Dj de la donnée à analyser et de la valeur "Val" de cette donnée extraite de la mémoire du terminal. Ces blocs sont mémorisés les uns à la suite des autres dans une zone de la mémoire du terminal appelée "ZDR". Le contenu de cette zone est envoyée vers le réseau à la fin de la transaction ou lors d'une demande par le réseau des données d'autodiagnostic. Une fois toutes les données transmises, la zone ZDR est vidée pour être re-utilisée lors d'une nouvelle introduction de carte.

Si Sk est égal à "2", la donnée doit être envoyée vers l'imprimante du terminal, le programme se poursuit par l'étape 11. Au cours de cette étape 11 un message est créé dans le tampon de l'imprimante, composé d'un texte (code ASCII) rappelant la nature de la donnée par exemple " MONTANT : " suivie de la valeur décimale ou hexadécimale de la donnée Dj, le message se termine par un séparateur et un " RETOUR CHARIOT - SAUT DE LIGNE ". Avantageusement, on peut regrouper tous les messages et les imprimer à la fin de la transaction.

Si Sk est égal à "3", la donnée doit être envoyée vers l'afficheur du terminal, le programme se poursuit par l'étape 12. Au cours de cette étape 12, un message est alors créé dans le tampon de l'afficheur, composé d'un texte (code ASCII) rappelant la nature de la donnée par exemple "MONTANT : " suivie de la valeur décimale ou hexadécimale de la donnée Dj. Les messages correspondant à chaque élément (Tt, Dj, "3") sont affichés

successivement pendant un certain temps fixé par le logiciel d'autodiagnostic. Avantageusement, on peut regrouper tous les messages et les afficher à la fin de la transaction, le défilement des messages peut être contrôlé par l'appui sur une touche du clavier du terminal.

- 5 Si Sk est égal à "4", la donnée doit être mémorisée dans la carte, le programme se poursuit par l'étape 13. Au cours de cette étape 13, un quatrième champ est réservé à cet usage, dans ZD, l'adresse "Adr-v" de ce quatrième champ et le nombre d'octets "Nb_v" de ce champ ont été mémorisés dans la zone ZTD lors du chargement des données
- 10 d'autodiagnostic dans le terminal. Le terminal envoie donc à la carte une commande d'écriture avec les paramètres suivants :

Adresse d'écriture : Adr-v

Nombre d'octets à écrire : Nb-v

Valeur à écrire : "Val" celle de la donnée Dj.

- 15 L'écriture dans la zone ZD n'est autorisée par le système d'exploitation de la carte que dans le quatrième champ d'un triplet de type Sk=4 correspondant à la Ti. Dans le cas où le triplet Ti de la carte n'est pas de type Sk=4, l'écriture est refusée. Un compte rendu d'exécution est systématiquement renvoyé au terminal après chaque commande d'écriture, si celle-ci n'a pas
- 20 été menée à bien, le terminal avertit l'utilisateur par un message. L'utilisation des données mémorisées est expliquée dans la suite. Une variante consiste à stocker temporairement toutes les valeurs des données Dj de type Sk=4 et à effectuer les commandes d'écriture des valeurs à la fin de la transaction.

- Une fois la donnée Dj traitée, à l'étape 14, le programme vérifie s'il existe
- 25 dans la zone ZTD d'autres triplets pour lesquels Tt = Ti. Si oui, le programme reboucle à l'étape 9 et traite un nouveau triplet. Pour chaque tâche élémentaire, la recherche de triplets s'effectue en balayant toute la zone ZTD. S'il ne reste plus de triplets (Ti=TT, Dj, Sk) à traiter, le programme passe à l'étape 15 en recherchant d'autre tâche à exécuter. S'il n'y a plus
- 30 d'autres tâches à exécuter, le programme reboucle à l'étape 2, en attente d'une nouvelle introduction de carte.

Dans le cas d'une carte banale utilisée pour transmettre les données d'autodiagnostic, la fonction d'autodiagnostic doit pouvoir s'exécuter qu'une seule fois. En effet, l'opérateur peut ne vouloir effectuer qu'un seul test du terminal lecteur de carte, ensuite les données ne doivent pas sortir du terminal. De plus, si la donnée à contrôler possède un champ Sk de type 4, un seul stockage n'est pas possible. Pour recommencer une nouvelle fois la fonction, la carte doit être de nouveau programmée par l'opérateur. Pour éviter d'être utilisée plusieurs fois avec la fonction d'autodiagnostic, juste après la lecture de la zone ZD par le terminal, le système d'exploitation de la carte va positionner à l'état inactif l'indicateur Ind_D. Pour des raisons de sécurité, il peut aussi effacer tous les triplets de type Sk=1, 2 et 3. L'indicateur Ind-D est inactif, la lecture de la zone ZD n'est plus possible.

Les données correspondant au type Sk=4 sont traitées lorsque la carte banale passe dans un terminal habilité à les lire, c'est à dire un terminal qui s'est authentifié de la même façon que lors de l'écriture des données d'autodiagnostic. Une fois tous les triplets lus, l'effacement total de la zone ZD peut être effectué. Cet effacement peut être provoqué par une commande spéciale ou lors de la première écriture dans la zone ZD. L'effacement, que justifie la sécurité, permet aussi de libérer la place occupée par la zone ZD. Cet emplacement peut être utilisé par l'application.

Ce fonctionnement style "mono-coup" de la fonction d'autodiagnostic peut être intéressant, lorsqu'une personne rapporte sa carte de crédit à une agence bancaire en déclarant que, sur un certain type de terminal de paiement, sa carte "ne marche pas". L'agence va enregistrer les données d'autodiagnostic dans cette carte en spécifiant que les données de la transaction (montant, date, valeur du certificat) seront écrites dans la carte en positionnant dans chaque triplet correspondant à une tâche à enregistrer le troisième champ Sk à la valeur "4" (Sk=4). La personne retourne chez le commerçant où se trouve le terminal à tester, effectue une transaction et revient à son agence qui analyse ou fait analyser à distance les informations stockées dans ZD.

Le fonctionnement mono-coup est aussi intéressant pour vérifier le fonctionnement d'un terminal soupçonné de fraude. Un organisme bancaire repère que, sur un terminal, des transactions sont créditées sans qu'il existe

des demandes de débit sur des comptes clients. L'organisme bancaire va envoyer des contrôleurs dotés de cartes banales avec la fonction d'autodiagnostic. A leurs retours, les données chargées dans la carte seront analysées.

- 5 Un autre exemple : un organisme bancaire peut trouver intéressant de connaître rapidement le moment et l'endroit où une carte est utilisée pour la première fois. Pour cela, avant de délivrer la carte à son porteur, cette dernière contient deux triplets de type $Sk=1$ dans lequel chaque troisième champ Sk est à la valeur "1". Dans la zone ZD, avec la donnée Dj correspondant à la date de la transaction et la donnée Dj correspondant à l'identité du terminal. Lors de la première transaction, les deux blocs (Ti , Dj , "date") et (Ti , Dj , "identité du terminal") seront envoyés immédiatement sur le réseau.
- 10

REVENDICATIONS

1. Terminal doté d'un programme d'application, d'au moins une sortie constitué soit par un affichage, soit par une imprimante, soit par un réseau de communication, soit par un objet portable et coopérant avec un objet portable 5 doté d'une zone de mémoire non-volatile (ZD) contenant des données et comportant un lecteur communiquant avec ledit objet portable, caractérisé en ce que l'appareil comporte des moyens de lecture ou de stockage dans sa mémoire de données (Ti, Dj, Sk) d'autodiagnostic ou de supervision et un moyen d'émission desdites données vers des sorties (1-4) 10 spécifiées en fonction d'informations fournies par les données d'autodiagnostic ou de supervision suite à l'exécution d'au moins une tâche Tt de son programme d'application en relation avec l'objet portable.
2. Terminal, selon la revendication 1, caractérisé en ce que les moyens d'émission des données d'autodiagnostic sont activés un certain nombre de 15 fois.
3. Terminal, selon la revendication 1, caractérisé en ce que les moyens d'émission des données d'autodiagnostic ou de supervision comporte des moyens d'écriture dans l'objet portable connecté à l'appareil.
4. Terminal, selon une des revendications 1 à 3, caractérisé en ce que les 20 données d'autodiagnostic ou de supervision sont constituées par au moins un triplet (Ti, Dj, Sk) d'information correspondant pour une première information (Ti) à une tâche déterminée du programme d'application, pour la seconde information (Dj) à un type de données corrélées à la tâche exécutée et à présenter à une sortie, et pour la troisième information à une 25 valeur (Sk) de spécification de la sortie à laquelle le type de données doit être présenté parmi celles présentes sur le terminal.
5. Terminal, selon une des revendications précédentes, caractérisé en ce que l'appareil dispose d'un moyen de test de la présence de données d'autodiagnostic ou de supervision dans un objet portable et de 30 déclenchement de la lecture et du stockage de ces données dans une zone spécifique ZTD de la mémoire du terminal.

6. Terminal, selon une des revendications précédentes, caractérisé en ce qu'il comporte des moyens d'introduction de données d'autodiagnostic ou de supervision dans un objet portable.

5 7. Procédé de supervision du fonctionnement d'un terminal ou d'autodiagnostic d'un terminal à partir d'un triplet d'information correspondant pour une première information à une tâche déterminée d'un programme d'application exécuté soit par un objet portable soit par un terminal, pour la seconde information à un type de données corrélées à la tâche exécutée et à présenter sur une sortie et pour la troisième information
10 à une valeur de spécification de la sortie parmi celles présentes sur le terminal caractérisé en ce qu'il consiste :

- à exécuter (9) une tâche du programme d'application sur le terminal ;
- à tester (9) un indicateur soit dans le terminal soit dans l'objet portable pour déterminer si une fonction d'autodiagnostic ou de supervision est
15 opérationnelle, puis dans le cas d'une réponse positive ;
- à rechercher (9) dans la mémoire soit de l'objet portable soit du terminal s'il existe parmi les triplets d'information stockés un triplet dont la première information correspond à la tâche déterminée exécutée par le terminal ou la carte ;
- 20 - à émettre vers la sortie spécifiée par le triplet ainsi lu la valeur de la donnée corrélée à la tâche exécutée et à référencer par la seconde information du triplet.

8. Procédé, selon la revendication 7, caractérisé en ce qu'il comporte une étape de test (16) consistant à déterminer s'il existe d'autres tâches à
25 exécuter et à la suite de l'exécution de ces tâches à rechercher tous les triplets d'information correspondants à l'exécution de cette tâche.

9. Procédé, selon une des revendications 7 et 8, caractérisé en ce qu'il comporte une étape de lecture d'un objet portable mémorisant dans sa mémoire non-volatile une pluralité de triplets et une étape (6) de stockage
30 de ces triplets dans une zone de mémoire non-volatile du terminal suivie

d'une étape (7) d'activation d'un indicateur de fonction d'autodiagnostic ou de supervision active.

10. Procédé, selon une des revendications 7 à 9, caractérisé en ce qu'il comporte une étape de test (3, 4) pour déterminer si l'objet portable est une
5 carte spécifique à la fonction d'autodiagnostic ou de supervision ou une carte dite banale.

11. Procédé, selon une des revendications 7 et 10, caractérisé en ce que les données d'autodiagnostic ou de supervision sont constituées par un quatrième champ d'information contenant au départ dans l'objet portable
10 l'adresse d'écriture (Adr-V), le nombre d'octets à écrire (Nb-V), après l'opération d'autodiagnostic la valeur à écrire (Val).

12. Objet portable utilisable avec le terminal selon une des revendications 1 à 6, caractérisé en ce qu'il comporte une carte à microprocesseur fonctionnant grâce à un système d'exploitation mémorisé sur la carte et
15 comportant une mémoire non-volatile contenant au moins un triplet (Ti, Dj, Sk) d'information dans une zone ZD déterminée de cette mémoire non-volatile dont la position est définie par des champs d'adresse situés dans la partie de mémoire servant à stocker le système d'exploitation.

13. Objet portable selon la revendication précédente, caractérisé en ce que
20 la partie (25) de mémoire non-volatile servant à mémoriser le système d'exploitation comporte également dans un champ mémoire une information constituant un compteur (21) d'utilisation de la fonction d'autodiagnostic.

14. Objet portable selon une des revendications 12 et 13, caractérisé en ce que la zone mémoire (25) stockant le système d'exploitation comporte un
25 champ permettant de mémoriser un indicateur (232) d'activation de la fonction d'autodiagnostic ou de supervision.

1/4

		D1 Valeur du code porteur	D2 Identité du porteur	D3 Valeur de la transaction	D4 Date de la transaction
T1	Vérification habilitation carte	SK	X SK=1		
T2	Authentification du porteur	X SK=2			
T3	Acquisition des données transaction			X SK=1	
T4	Validation de la transaction			X SK=3	X SK=1

Fig.1

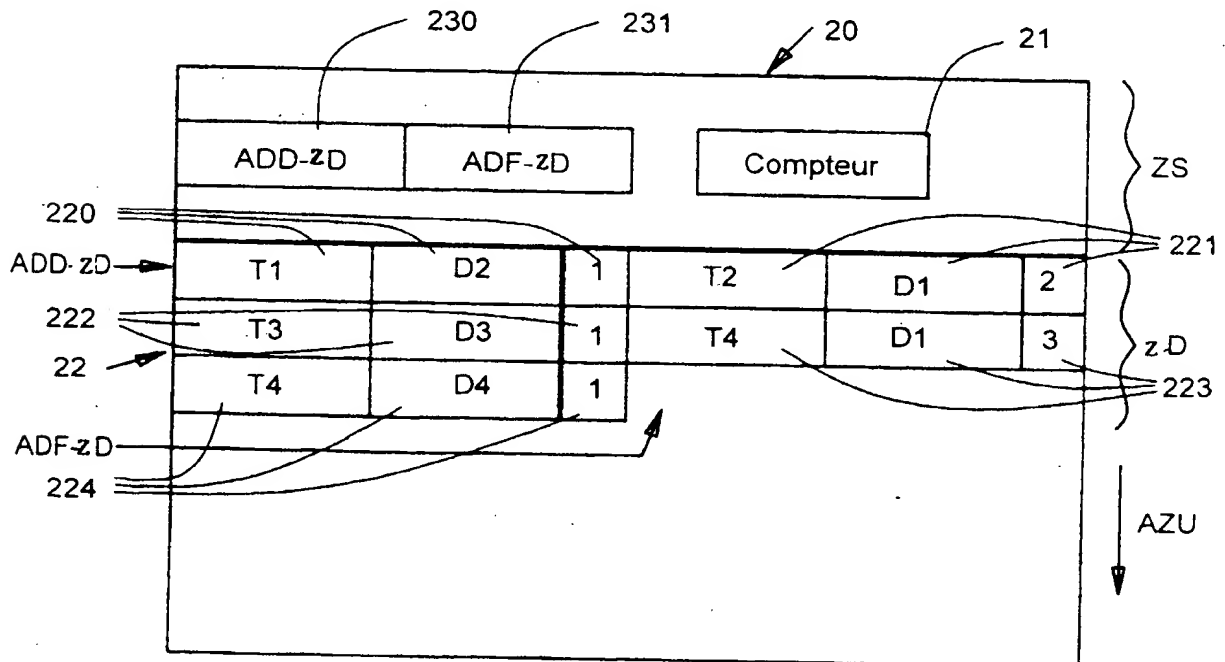
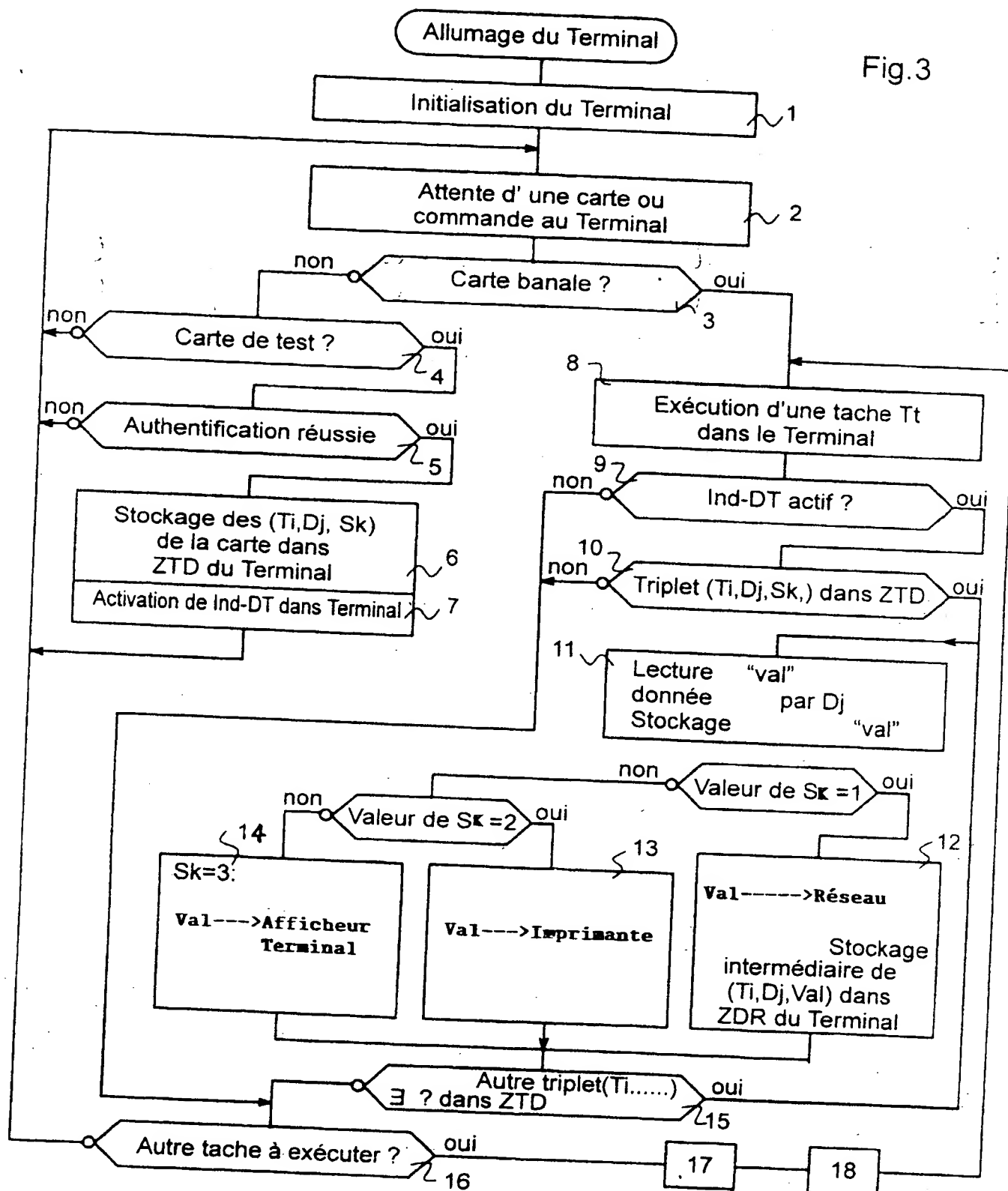


Fig.2

Fig.3



3/4

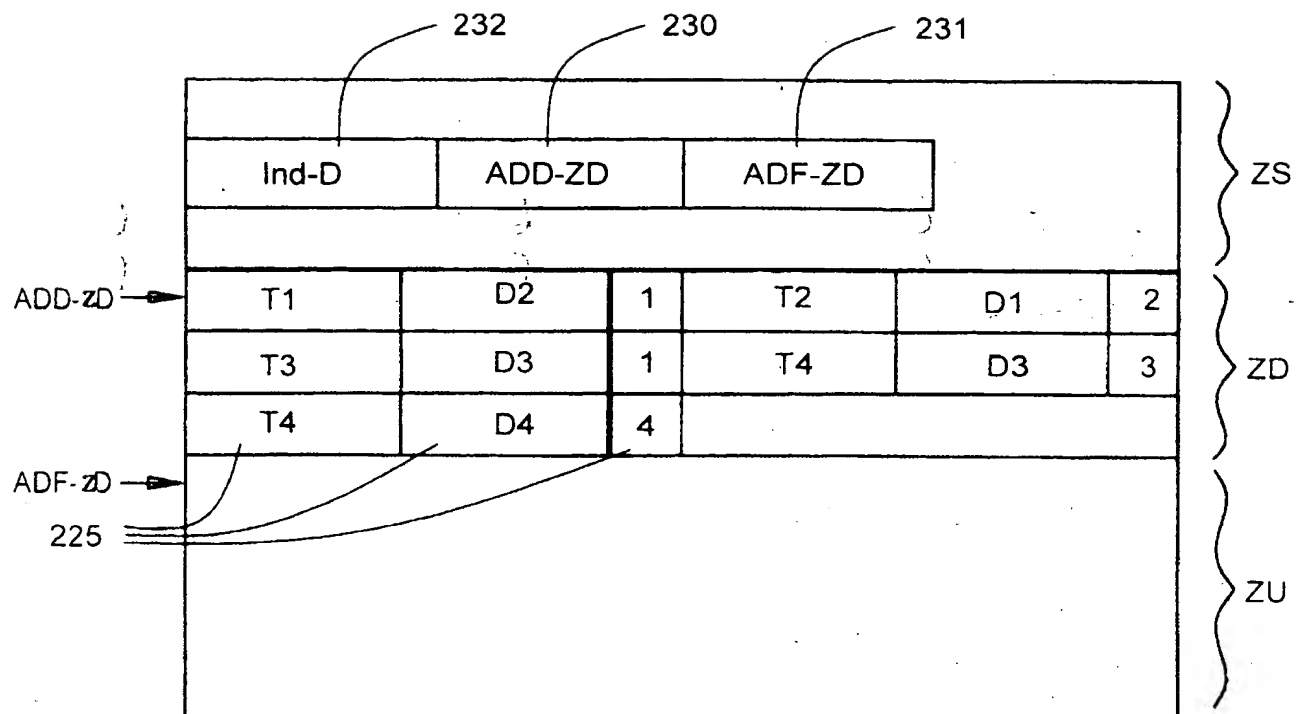
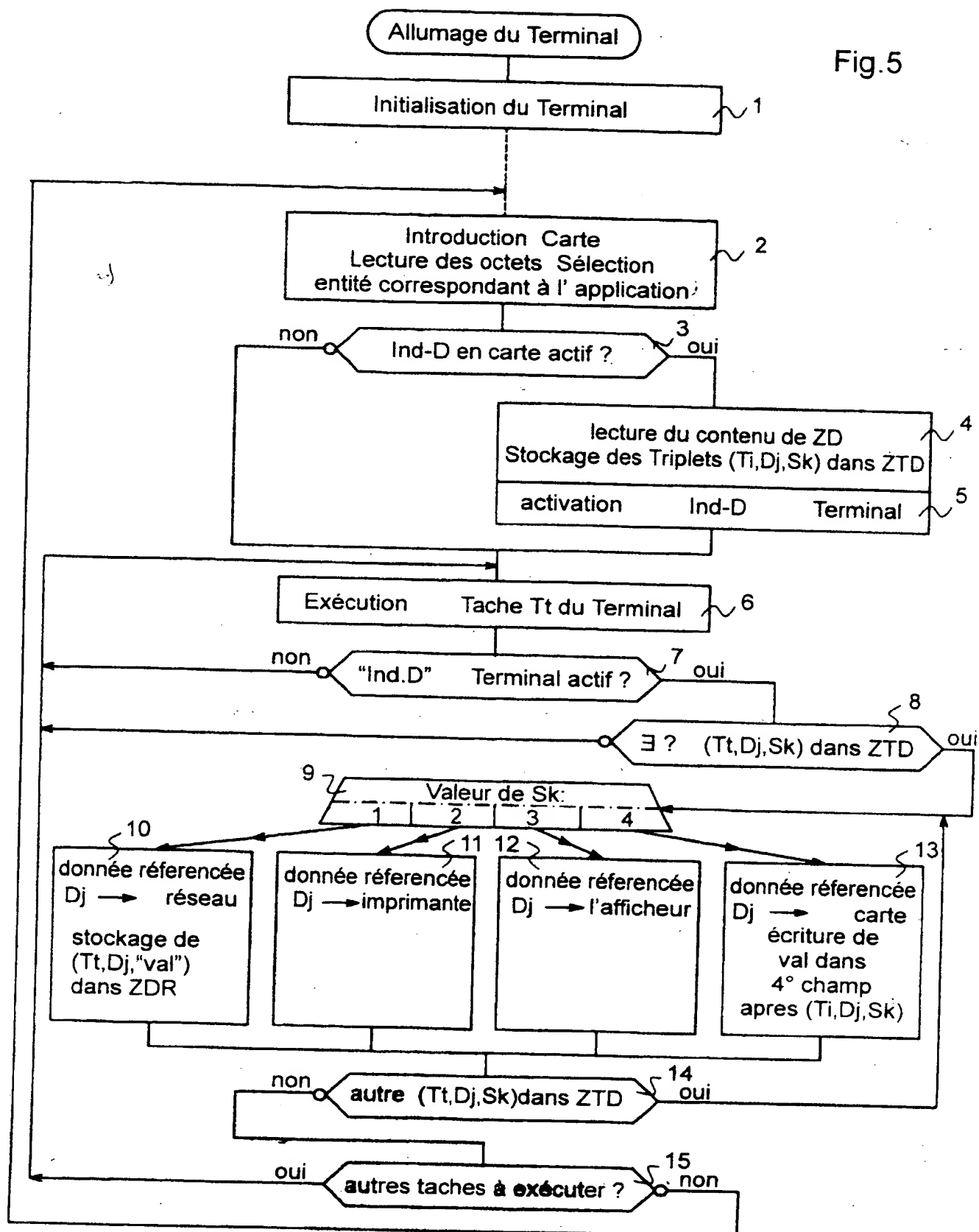


Fig.4

4/4

Fig.5



INSTITUT NATIONAL
de la
PROPRIETE INDUSTRIELLE

RAPPORT DE RECHERCHE
PRELIMINAIRE

établi sur la base des dernières revendications
déposées avant le commencement de la recherche

N° d'enregistrement
national

FA 540700
FR 9615997

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
A	EP 0 387 972 A (KLÜSSENDORF) * le document en entier *	1,3,6,7, 9,12
A	EP 0 131 331 A (MAINMET) * abrégé; revendications; figures 1,2 * * page 4, ligne 19 - page 5, ligne 37 * * page 7, ligne 10 - ligne 30 *	1,7,10, 12
A	US 4 777 355 A (K. TAKAHIRA)	
		DOMAINES TECHNIQUES RECHERCHES (Int.CL.6)
		G07F
Date d'achèvement de la recherche		Examineur
23 octobre 1997		David, J
CATEGORIE DES DOCUMENTS CITES		
X : particulièrement pertinent à lui seul Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général O : divulgation non-écrite P : document intermédiaire		
T : théorie ou principe à la base de l'invention E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure. D : cité dans la demande L : cité pour d'autres raisons & : membre de la même famille, document correspondant		

1

EPO FORM 1503 03.82 (P04C13)

)))
)))
THIS PAGE BLANK (USPTO)